














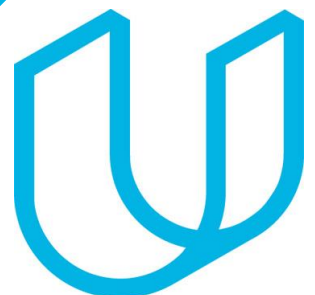


TimeSheets: Threat Report

**YOUR NAME : Shrividya Ranjani Kaliyur
NarayanaPrasad
DATE : 03/22/2021**



Purpose of this Report:

This is a threat model report for **TimeSheets**. The report will describe the threats facing TimeSheets. The model will cover the following:

- Threat Assessment
 - Scoping out Asset Inventory
 - Architecture Audit
 - Threat Model Diagram
 - Threats to the Organization
 - Identifying Threat Actors
- Vulnerability Analysis
- Risk Analysis
- Mitigation Plan



Section 1

Initial Threat Assessment

Completed Asset Inventory

Components and Functions

- ***TimeSheets Web Server:*** The web server's primary role is to serve static content to a requesting client through the http protocol.
- ***TimeSheets Application Server:*** The application server handles all the business logic process and serves dynamic content.
- ***TimeSheetsDB:*** The database server stores employee data and will be queried from the application server.
- ***AuthDB:*** Stores user authentication data (credentials) and will be queried from the application server.

Completed Asset Inventory

Overview of Application Functionality

TimeSheets is used by employees to track their hours worked. Users will login to the TimeSheets application from their device.

Data Flow

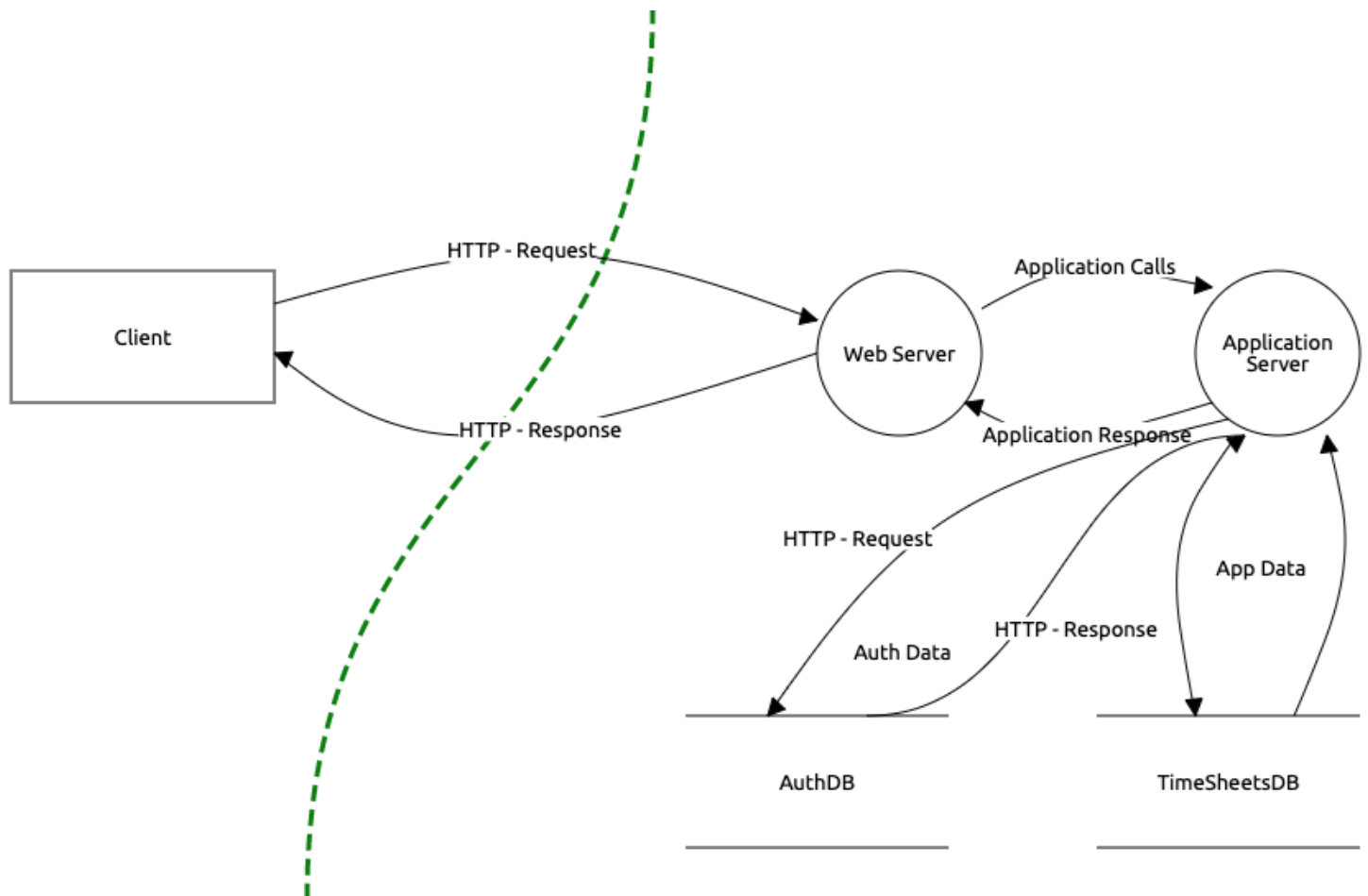
Request is generated from the client via the Internet. The request arrives at the TimeSheets web server which serves static content to the user (HTML, images, etc). Dynamic data is retrieved from the database and served to the client.

Completed Architecture Audit

Flaws

- *There is a lack of encryption at rest - database servers are storing data on unencrypted disks.*
- *There is lack of redundancy.*
- *There is no firewall that is filtering traffic coming from the Internet*

Completed Threat Model



- Employee Data Unencrypted at Rest
- Authentication data is using reversible encryption
- Authentication requests are not encrypted in transit
- Sensitive data is encrypted using DES algorithm

Completed Threat Analysis

What Type of Attack Caused the Login Alerts?

Man in the Middle (MitM)

What Proves Your Theory?

There is lack of encryption between the client and the application. A malicious actor is sniffing traffic and intercepting the requests with a valid username/password in the request. Additionally, the logs show successful login attempts from the expected IP, but also a different location at the same time.

Completed Threat Actor Analysis

Who is the Most Likely Threat Actor?

Internal User

What Proves Your Theory?

The IP address of the unexpected login matches that of an internal user. Additionally, there was no data leaked from the company, and no data changed. Just data accessed that the legitimate user typically doesn't access.



Section 2

Vulnerability Analysis

2.1 Employee Data Unencrypted at Rest

Discovery:

During the threat model the SRE team confirmed that the database is on a server that does not have encryption at rest.

Why is this an issue?

Employee data may contain employee name, address, phone number and SSN number.

1. Hackers can stay on the servers without detection by using the employee data like username and password to login and spy on the company for a very long time and get to know all the company secrets.
2. If a third person gets access to all the employee data, then it breaks the employee privacy policy.
3. There is a risk of price drop of the company stocks as no one will want to invest in a company who couldn't protect their employee data.

2.2 Authentication Data Stored Using Reversible Encryption

Discovery:

During the threat model the DBA team confirmed that the database is storing authentication data (credentials) encrypted.

Why is this an issue?

1. If the encryption being used on to store authentication data can be reversible, then it becomes nechter to hackers.
2. Hackers can get access to all the authentication factors that are used to prove the authentication and authorization of a person.
3. If an outsider get their hands on authentication data, they could use it to get access to the company systems and data.
4. Once the hackers get access to the company data they can modify, delete or sell the company data in the black market or block access and demand for ransom.
5. Hackers can impersonate themselves as an authenticated personnal.

2.3 Authentication Requests are Unencrypted in Transit

Discovery:

During the threat model the security team confirmed that authentication requests are being transmitted in plaintext.

Why is this an issue?

The following are the risks associated to unencrypted authentication requests in transit:

1. Eavesdropping : A hacker might capture packets from the requests that are in transit, via computers or other devices. These captured packets can reveal all the authentication requests to the hacker.
2. Man-in-the-Middle : A hacker can secretly attack, alter the communication between the two parties or devices. By doing this the hacker will get access to all the communications, requests and authentication factors.

2.DES Algorithm in Use

Discovery:

During the threat model the security team identified sensitive data being stored using the DES algorithm.

Why is this an issue?

The following are the risks associated to DES algorithm:

1. Brute Force Attack : Since DES algorithm has a 64-bit key, the number of combinations are small, and a simple personal computer can break it.
2. When two inputs are given to an S-box they can create the same output.
3. Since DES algorithm is outdated, techniques to exploit the algorithm might be known and could be easily exploited.

Optional Task:

Examine the threat model diagram from Section 1 and answer:

What non-encryption issues can you identify?

What recommendation would you give to solve those issues?

Why do you recommend those solutions?

- *[Issue 1 Here]*
- *[Issue 2 Here]*
- *[Add more issues as necessary]*



Section 3

Risk Analysis

3.1 Scoring Risks

Risk	Score <i>(1 is most dangerous, 4 is least dangerous)</i>
Unencrypted at Rest	3
Reversible Encryption	4
Unencrypted in Transit	1
Outdated Algorithm	2

3.2 Risk Rationale

Why Did You Choose That Ranking? Make sure to include your risk ranking methodology.

The ranking of the given risks was done according to a basic research from Google.

- Unencrypted authentication requests via transit is ranked 1 since attacks like Man-in-the-Middle can allow the attacker to get all the information that wasn't intended for them to get access to. Important information like username and password could be accessed by the attacker. The likelihood of this risk is high.
- Outdated algorithm was ranked 2 since the likelihood and impact of the risk is high. Weaknesses of outdated algorithms are known to everyone and attackers can use those weaknesses to create new and creative ways to get into the system or gain access to information.
- Unencrypted employee data is ranked 3 since the likelihood of an attack of this medium but the impact might be a little less than the other risks.
- Reversible encryption in use is ranked 4 since the likelihood of an attacker finding out that the encryption is reversible is low.

The method that was used to calculate the risk assessment was:

$\text{Risk} = \text{Likelihood} * \text{Impact}$



Section 4

Mitigation Plan

4.1 Employee Data Unencrypted at Rest

What is Your Recommended Mitigation Plan?

Use of Transparent data encryption and encrypt the disks

Why Did you Recommend This Course of Action?

TDE is used to encrypt SQL server and Azure SQL Database data and files at rest. It performs a real time I/O encryption and decryption of the data and log files to protect the data at rest.

Encrypting a disk makes sure that even if the system gets into the wrong hands, they won't be able to access the data.

The steps to follow to encrypt a disk are:

1. Turn on BitLocker for the selected disk.
2. The disk can either be encrypted by using a USB flash drive or a password.
3. Save the copy of the recovery key somewhere safe.
4. Chose how much of the disk to encrypt.
5. Chose which encryption mode to use.
6. Encrypt the disk.

4.2 Authentication Data Stored Using Reversible Encryption

What is Your Recommended Mitigation Plan?

Set Store password using reversible encryption to disable and use salted hash.

Why Did you Recommend This Course of Action?

By setting Store password using reversible encryption to enabled, it makes it possible to decrypt passwords. But an attacker can use this to log in to the network resources. By setting Store password using reversible encryption to disable will decrease the chance of an attack.

Use of normal hashes can leave the content vulnerable to dictionary attacks. Therefore, salted hash can be used to store authentication data to protect the content from dictionary attack.

Steps to implement Salted Hash:

1. Take a password and a salt of your choice.
2. Append or prepend the salt to the password. This becomes the salted password.
3. Hash the salted password using a hashing algorithm.

4.3 Authentication Requests are Not Encrypted in Transit

What is Your Recommended Mitigation Plan?

End to end encryption

Why Did you Recommend This Course of Action?

End-to-end encryption ensures that the data is protected, not modified and understood by the actual receiver. The data in transit is encoded to prevent anyone from being able to read it. The data is kept encrypted until it reaches the destination.

4.4 DES Algorithm in Use

What is your recommended Mitigation Plan?

Use of AES

Why did you recommend this course of action?

AES known as Advanced Encryption Standard is based on the principle of substitution and permutation. DES uses plain text of 64-bits only whereas, in AES plaintext can be 128, 192, 256 bits.

AES is much more secure than DES and is widely used.

4.5 Security Audit

The audit team has been made aware of the systemic issue and wants to ensure your recommendations are followed. What steps can the audit team take?

The following steps are to be followed by the audit team:

1. Create a policy that contains all the steps mentioned below:

- a) Establishment of a Transparent Data Encryption on the employee at rest.
- b) Set Store password using reversible encryption to disable.
- c) Use of Salted hashing.
- d) Establish end to end encryption.
- e) Use AES algorithm instead of DES algorithm.

2. Maintain a detailed documentation of the procedures used to implement the recommendations. Maintaining a documentation makes it easy for a person to analyze what when wrong if something fails. A new employee who joins the company in the future will have all the materials required to understand what is done and what is being done.

3. Conduct a recurring review on all the systems regularly. Conducting recurring internal audits ensures compliance.

When a new policy is created it must be communicated with the team as it is important that everyone is aware of what is being done and what should be done. This reduces the confusion between the team and reduces a chance of failure.

Optional Task:

Create an architecture diagram of a secure system.

Image of your secure architecture:

Optional Task (*Continued*):

Additional Steps Would You Recommend to Prevent the Attack as well as Future Issues: